International Journal of Computer Science and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 4, Issue 1, Feb 2013, 1-6

TRANS STELLAR
• Journal Publications • Research Consultancy

PERFORMANCE EVALUATION OF DATA CLASSIFICATION USING CONVENTIONAL CLASSIFIERS

S. V. S. GANGA DEVI
Professor in MCA, K. S. R. M. College of Engineering, Kadapa, Andhra Pradesh, India

ABSTRACT

Accuracy of data classification strongly depends on the dataset used for learning. High dimensional datasets often suffer from the 'Curse of dimensionality' problem which degrades the accuracy and performance of the classification models. Feature selection is an important step in building an effective and efficient classifier. In this paper, performances for Eucalyptus and Fruit data classification of popular classifiers are compared qualitatively.

KEYWORDS: Data Mining, Classification, Performance Evaluation

INTRODUCTION

Data classification is an important task in KDD (Knowledge Discovery in Databases) process [15]. It has several potential applications. The performance of a classifier is strongly dependent on the dataset used for learning. In practice, a data set may contain noisy or redundant data items and a large number of features and many of them may not be relevant for the objective function at hand. Thus high dimensional data sets often suffer from the curse of dimensionality problem which degrades the accuracy and performance of the classification models. Thus, feature selection is an important step in building an effective and efficient classifier. It is the process that chooses an optimal subset of features according to an objective function. Feature selection reduces dimensions and simplifies the data. It leads to better performance of the classification models in terms of their predictive or descriptive accuracy, diminishing computing time needed to build models as they learn faster and better understanding of the models [10]

In this paper, a comparative analysis of classification for Eucalyptus soil conservation data and Fruit data are presented. The predictive performances of popular classifiers are compared quantitatively. The rest of the paper is structured as follows. The following section presents the literature review on conventional classifiers. Section 3 describes the research approach of the experiment taken and the results of study and finally section 4 ends with conclusion.

LITERATURE REVIEW

Decision Tree

A decision tree partitions the input space of a dataset into mutually exclusive regions, each of which is assigned a label, a value or an action to characterize its data points. The decision tree mechanism is transparent and one can follow a tree structure easily to see how the decision is made. A decision tree contains internal and external nodes connected by branches. An internal node is decision-making unit that evaluates a decision function to determine which child node to visit next. The external node, on the other hand, has no child nodes and is associated with a label or value that characterizes the given data being visited. However, many decision tree construction algorithms involve a 2-step process. First, a very large decision tree is grown. The given tree is pruned to reduce large size and over-fitting the data in the second step. The pruned Decision tree that is used for classification purposes is called the classification tree. A popular decision tree algorithm is C4.5. It can help not only to make accurate prediction from the data but also to explain the patterns in it. It deals with the

problems of the numeric attributes, missing values, pruning, estimating error rates, complexity of decision tree induction and generating rules from trees (Witten and Frank, 1999). In terms of predictive accuracy, C4.5 performs slightly better than CART and ID3 algorithms (Juan et al., 2011). The learning and classification steps C4.5 are generally fast (Han and Kamber, 2001). However, scalability and efficiency problems such as the substantial decrease in performance and poor use of available system resources can occur when C4.5 is applied to large datasets.

Boosting

Boosting [8][11]is the technique to improve the accuracy of the weak classifier. The idea is to generate several classifiers rather than one. And each classifier tries to classify accurately the classes, which were misclassified by the previous classifier. But how can we generate several classifiers from a single data set? As the first step, a single decision tree is constructed from the training data. This classifier will usually make mistakes on some cases in the training set. When the second classifier is constructed more attention is paid to these cases in an attempt to get them right. As a consequence, the second classifier will generally be different from the first. It also will make errors on some cases and these errors become the focus of attention during construction of the third classifier. This process continues for a predetermined number of iterations or trails, but stops if the most recent classifiers are either extremely accurate or inaccurate.

This improvement is achieved by giving weight to the individual elements of the training data set. The weights are initially set to be equal. Comparison is done to identify those elements of the training set which have been misclassified. These misclassified training data elements are then given an increased weight, and the classifier is run again. Increased weight of the misclassified elements forces classifier to focus on these cases.

Bagging

Bagging [3] is used to improve the classification model in terms of stability and classification accuracy. It also reduces variance and helps to avoid over fitting. Although it is usually applied to Decision Tree models, it can be used with any type of model also. Bagging is a special case of model averaging approach.

Given a set D of d tuples, Bagging works as follows. For iteration i (i=1,2, w), a training set D_i of d tuples is sampled with replacement from the original set of tuples D. The term Bagging stands for bootstrap aggregation. Each training set is a bootstrap sample. Because sampling with replacement is used, some of the original tuples of D may not be included in D_i, where as others may occur more than once. A classifier model, M_i is learned for each training set, D_i. To classify an unknown tuple X, each classifier M_i, returns its prediction, which counts as one vote. The bagged classifier M*, counts the votes and assigns the class with the most votes to X. The bagged classifier often has significantly greater accuracy than a single classifier derived from D, the original trained data. It will not be considerably worse and is more robust to the effects of noisy data. The increased accuracy occurs because the composite model reduces the variance of the individual classifiers.

Neural Network

Back-Propagation (BP) neural networks can process a very large number of instances, have a high tolerance to noisy data, and has the ability to classify patterns for which they have not been trained (Han and Kamber, 2001). They are an appropriate choice of the results of the model are more important than understood (berry and Linoff, 2000). However, the BP algorithm requires long training time and extensive testing and retraining of parameters, such as the number of hidden neurons, learning rate and momentum to determine the best performance (Bigus, 1996).

Bayesian Networks

The classifier is a powerful probabilistic representation and its use for classification has received considerable attention. This classifier learns from training data, which is the conditional probability of each attribute A_i given the class label C. Classification is then done by applying Baye's rule to compute the probability of C given the particular instances of $A_1 \ldots\ldots A_n$ and then predicting the class with the highest posterior probability. The goal of classification is to correctly predict the value of a designated discrete class variable given a vector of predictors or attributes. In particular, the naïve Baye's classifier is a Bayesian network, where the class has no parents and each attribute has the class as its sole parent. Although the Naïve Bayesian (NB) algorithm is simple, it is very effective in many real-world datasets because it can give better predictive accuracy than well-known methods like C4.5 and BP (Domingos and Pazzani, 1996; and Elkan, 2001). However, when attributes are redundant and not normally distributed, the predictive accuracy is reduced (Witten and Frank, 1999).

K-Nearest Neighbor

The classifier is considered as a statistical learning algorithm. It is extremely simple to implement and leaves itself open to a wide variety of variations. The training potion of nearest-neighbor does little more than store the data points presented to it. When asked to make a prediction about an unknown point, the nearest-neighbor classifier finds the closest training-point to the unknown point and predicts the category of the training point according to some distance metric. The distance metric used in nearest neighbor methods for numerical attributes can be simple Euclidean distance.

EXPERIMENT AND RESULTS OF THE STUDY

This section describes the experiment and the Eucalyptus dataset and Fruit data set are used in the study. In Eucalyptus soil conservation data, the objective is to determine which seed lots are best for soil conservation. Eucalyptus data set contains 10 attribute variables and one class variable, totally 11 attributes. In this, 637 instances are used.

The Attributes are (1) Rainfall (mm per annum) – integer, (2) Seed lot number – integer (PMCno), (3) Best diameter base height (cm) – real (DBH), (4) Height (m) – real (Ht), (5) Survival – integer (Surv), (6) Vigour – real (Vig), (7) Insect resistance – real (Ins_res), (8) Stem form – real (Stem_Fm), (9) Crown form – real (Crown_Fm), (10) Branch form – real (Branch_Fm). The class utility belongs to either of the classes none, low, average, good and best.

The fruit transported takes weeks to reach the market. In Fruit data, the goal is to determine which pre-harvest variables contribute to good tasting fruit after different periods of storage time. This is determined by whether a measure of acceptability found by categorizing each fruit as either not_acceptable, ok (acceptable) or excellent. Fruit data set contains 10 attribute variables and one class variable the Attributes are (1) Number of days after flowering – integer (daf), (2) Weight of whole fruit in grams – integer (weight), (3) Weight of fruit after storage – integer (storewt), (4) Penetrometer indicates maturity of fruit at harvest – real (pene), (5) Solids – a test for dry matter – real, (6) Brix – A refactometer measurement used to indicate sweetness or ripeness of fruit – real, (7) Glucose – measured in mg/100g of fresh weight – real, (8) Fructose – measured in mg/100g of fresh weight – real, (9) Sucrose – measured in mg/100g of fresh weight – real, (10) Flavour – the mean of eight taste panel scores out of 1500 – real.

The class variable acceptability belongs to either of the classes excellent, ok (acceptable) and not_acceptable. 53 instances are used for fruit data set.

66% of data has been randomly chosen to be training set and the rest of the data are reserved for testing. To evaluate the performance evaluation of conventional classifiers, "WEKA" is used and the results are shown in Table 1 for Eucalyptus data and in Table 2 for Fruit data. In that the classifier accuracies and time taken to build the model and size if it is appropriate, are shown. J48 is the Weka version of C4.5 REP tree is called as Reduced Error Pruning Tree. MLP stands for Multi Layer Perceptron and Ib_k is the weka version for K-nearest neighbor.

Table 1: Performance Evaluation of Conventional Classifiers for Eucalyptus Data

Name of the Classifier	Correctly Classified	Incorrectly Classified	Time (Sec.)	Size
J48	59.63	40.37	0.19	163
REP Tree	60.09	39.91	0.09	61
Boosting	57.80	42.20	0.23	N/A
Bagging	58.26	41.74	0.19	N/A
MLP	59.17	40.83	4.50	N/A
Naïve Bayes	56.42	43.58	0.02	N/A
Bayes Net	53.20	46.79	0.06	N/A
IB_k	55.05	44.95	0.00	N/A

Table 2: Performance Evaluation of Conventional Classifiers for Fruit Data

Name of the Classifier	Correctly Classified	Incorrectly Classified	Time (Sec.)	Size
J48	68.42	31.58	0.08	16
REP Tree	52.63	47.37	0.02	07
Boosting	78.95	21.05	0.05	N/A
Bagging	63.16	36.84	0.03	N/A
MLP	73.68	26.32	0.50	N/A
Naïve Bayes	57.89	42.11	0.02	N/A
Bayes Net	63.16	36.84	0.02	N/A
IB_k	78.95	21.05	0.00	N/A

Here N/A stands for "Not Applicable", for example in MLP, the model size is specified by the model builder and is not something that can be evolved during model building.

CONCLUSIONS

In this paper, an experiment to find out predictive performance of different classifiers for Eucalyptus and Fruit data are described. For this, various conventional classifiers are selected and the performance was measured.

REFERENCES

1. Berry m and Lin off G (2000), Mastering Data Mining: The Art and Science of Customer Relationship management, John Wiley & Sons, New York, USA.
2. Bigus J (1996), Data mining with Neural Networks, McGraw Hill, New York, USA
3. Breiman L, (1996) "Bagging Predictors", *Machine Learning*, Vol.24 (2), pp.123-140.
4. Domingos P and Pazzani M (1996), "Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier", In Proceedings of the 13th Conference on Machine Learning, pp.105-112, Bari, Italy.
5. Elkan C (1997), Naïve Bayesian Learning, Technical Report CS97-557, Department of Computer Science and Engineering, University of California, San Diego, USA.

6. Elkan C (2001), "Magical Thinking in Data Mining: Lessons from Coil Challenge 2000", Department of Computer Science and Engineering, University of California, San Diego, USA.
7. Han J and Kamber M (2001), Data Mining Concepts and Techniques, Morgan Kaufmann.
8. Kearns M and Mansour Y, (1996) "On the Boosting ability of Top-down Decision Tree Learning algorithms", *Proceedings of the Twenty-eighth ACM Symposium on the Theory of Computing*, ACM Press: New York, NY.
9. Klosgen W and ZytkowJ M (2002), Handbook of Data Mining and Knowledge Discovery, OUP, Oxford.
10. Liu H (1998), Feature Extraction, Construction and Selection: A data Mining Perspective, ISBN 0-7923-8196-3, Kluwer Academic Publishers.
11. Quinlan J.R, (1996) "Bagging, Boosting and C4.5", *Proceedings of Eighteenth American Association for Artificial Intelligence*, pp 725-730, AAAI Press, Menlo Park, CA.
12. Witten I and Frank E(1999), Data Mining: Practical Machine Learning Tools and Techniques with Java, Morgan Kauffman Publishers, California, USA
13. Xing E, Jordan M and Karp R (2001), "Feature Selection for High-Dimensional Genomic Microarray Data", in the proceedings of the 15 International Conference on Machine Learning, pp.601-608.
14. Yang Y and Pederson J O (1997), "A Comparative Study on Feature Selection in Text Categorization", Proc. 14th International Conference Machine Learning, pp.412-420.
15. Yu L and Liu H (2004), "Redundancy Based Feature Selection for Microarray Data", in the Proceeding of the 10th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.